DoD LVC Architecture Roadmap (LVCAR) Study Status

Ken Goad
LVCAR Project Manager
USJFCOM

UNCLASSIFIED

The LVC Architecture Issue

- **Current LVC environments are not inherently interoperable.**
  - High Level Architecture (**HLA**) and Distributed Interactive Simulation (DIS) are most often used for integrating virtual and constructive assets,
  - Test & Training Enabling Architecture (**TENA**) is widely used in testing and to integrate live assets into exercises/events.
  - Common Training Instrumentation Architecture (**CTIA**) promotes commonality among the U.S. Army's instrumented ranges and home stations; LVC - Integrated Architecture (**LVC-IA**) is next-generation Army multi-echelon, integrated, joint, training and mission rehearsal environment.
- **Multiple protocols, gateways, and object models are often used to bring an LVC Environment together.**
  - Interoperability and efficiency issues arise when bringing disparate protocols and entities together in a common operational environment.
  - Complexity, disconnects, duplication of effort, risk, and costs increase with multiple architectures.

At least four communities agree; critical review needed to develop way forward for efficient, effective interoperability.

What We Are Doing

- **Developing a recommended "way ahead" regarding LVC interoperability across three broad areas of concern:**
  - Desired future technical architecture(s)
  - Desired business model(s)
  - Manner in which standards should be evolved and compliance evaluated

- **The "way ahead" will provide:**
  - Rationale for recommendations, citing the findings on which they are based
  - Recommendations on the required management / governance structures and processes to implement the "way ahead"
  - Recommended next steps (e.g., prototyping any new architecture)

Desired Outcomes / Effects

- **Achieve greater LVC interoperability**
- **More efficient federation composition and federate re-use**
- **Reduce / avoid duplication of efforts / costs**
- **Responsive to evolving requirements**
- **Maintain or increase innovation**
- **Achieve the network effect**
- **Address the needs of broadest user domain feasible (flexibility vs. cost vs. performance)**

Deliverables

- ✓ Project Plan
- ✓ Workshop #1 Report
- ✓ Literature Review Report
- ✓ Capabilities and Limitations of Current LVC Architectures
- ✓ Use Case Template
- ✓ Ideal Use Case Set
- ✓ Unified LVC Use Case Document
- ✓ Workshop #2 Report
- ✓ Use Case Requirements
- ✓ Capabilities and Limitations vs User Requirements Map
- ✓ LVCAR White Paper
- ✓ LVCAR Functional Requirements Document
- ✓ Capabilities and Limitations Unified Document
- ✓ Capabilities and Limitations vs Requirements Document
- ❍ Interim Report
- ❑ Business Model Comparison Document
- ❑ Standards Management and Evolution Process Model Comparison Document
- ❑ Alternatives Ranking Report
- ❑ Final Report

✓ Completed

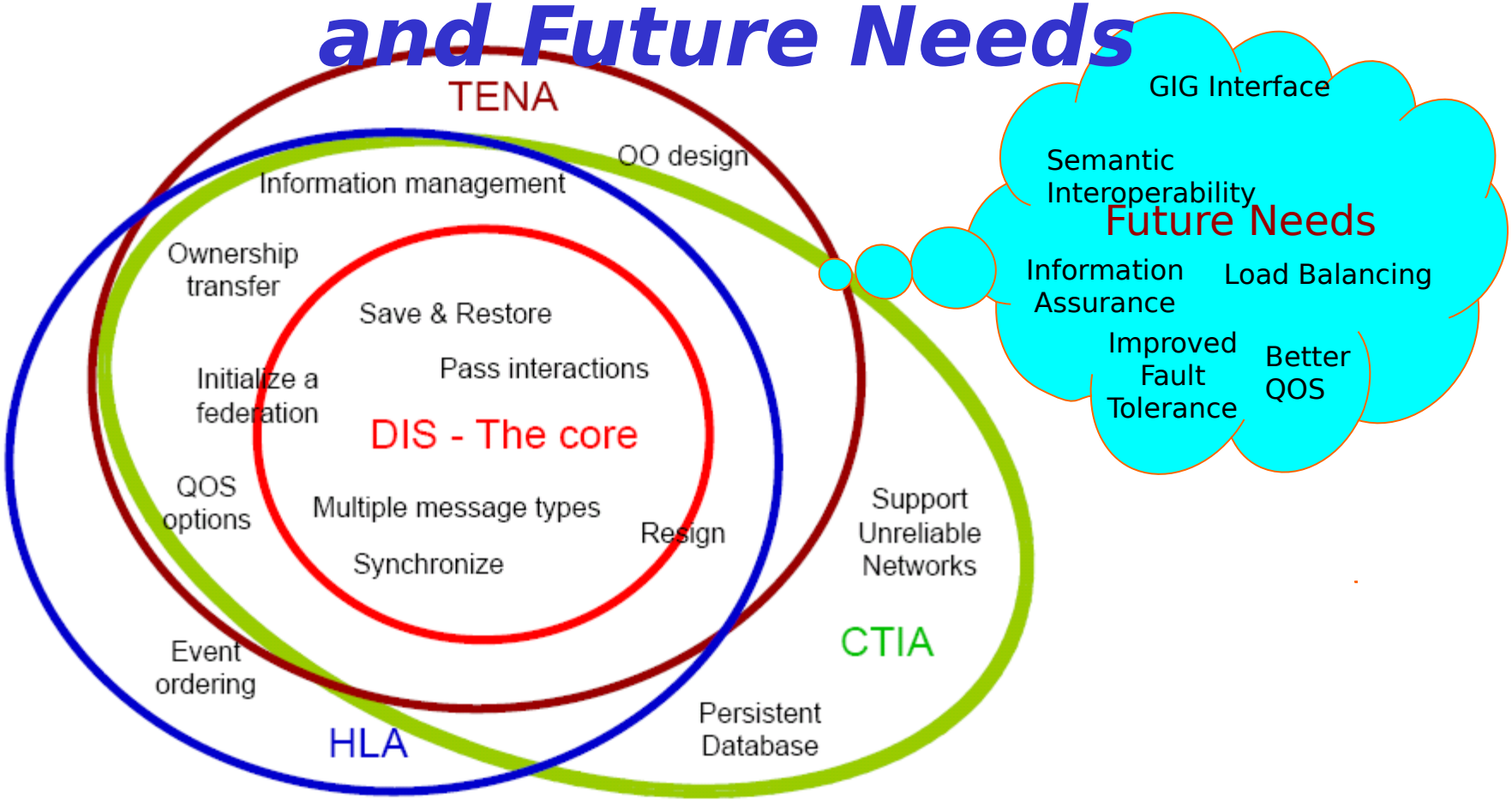
Requirements Data Sources

- **Foundational Documents (Existing Requirements)**
- **Workshop Grass-roots**
- **Survey Data**
- **RFIs as follow-on to Survey Data**
- **Use Cases**
- **Expert Team, Government Team, and Working Group Reviews**

Use Cases

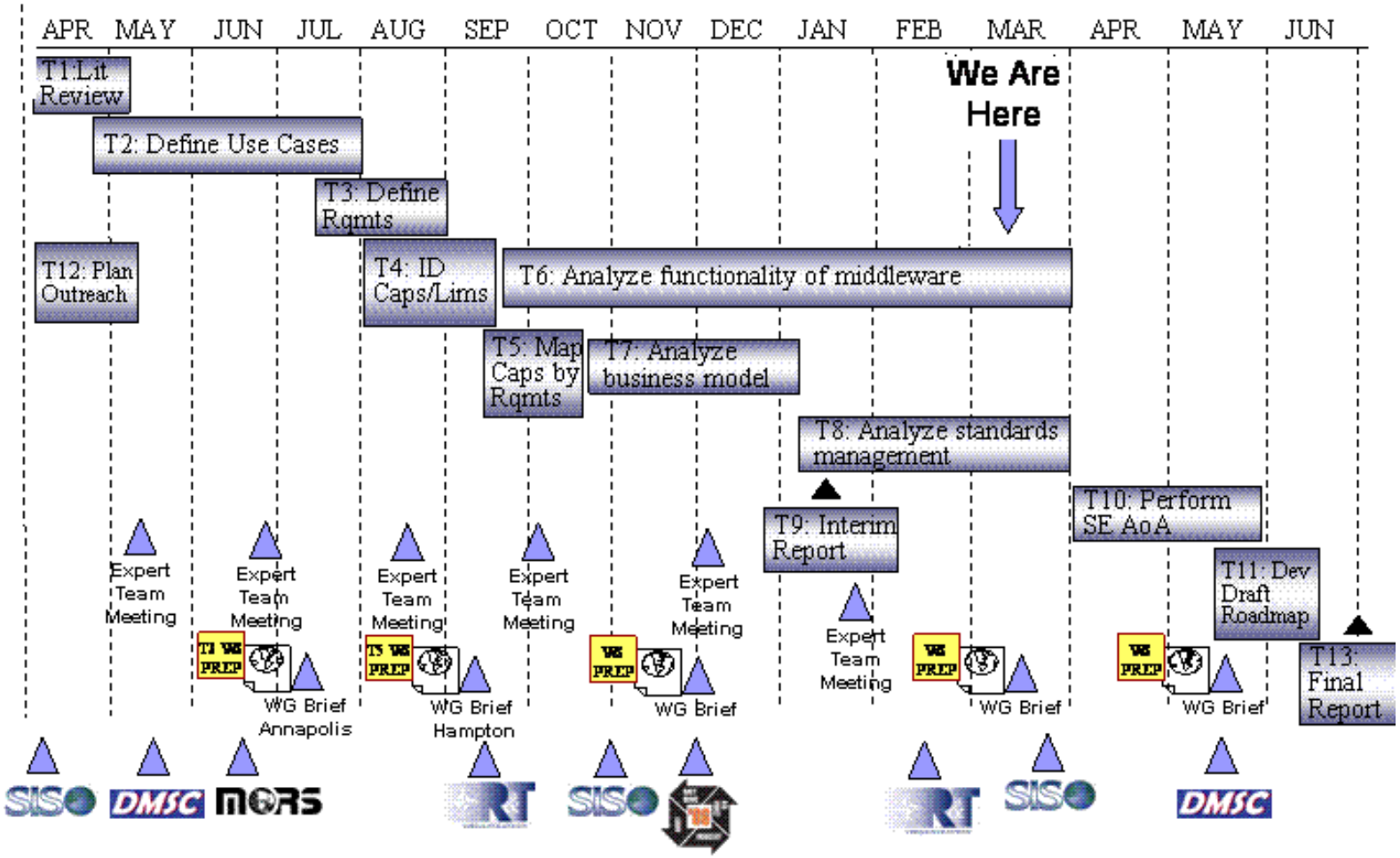
- Urban Resolve 2015
- DDG 1000 Design and Testing
- AF LVC Operations
- AVCATT and CCTT Interoperability
- JTEM Sys Eng
- ISR LVC Integration w/ Red Flag
- Heavy Brigade Combat Team
- Ulchi Focus Lens using ALSP
- Korean Battle Simulation Center
- NASA
- FCS Imbedded Training
- CVN-21

Integrating Architecture Overlap and Future Needs

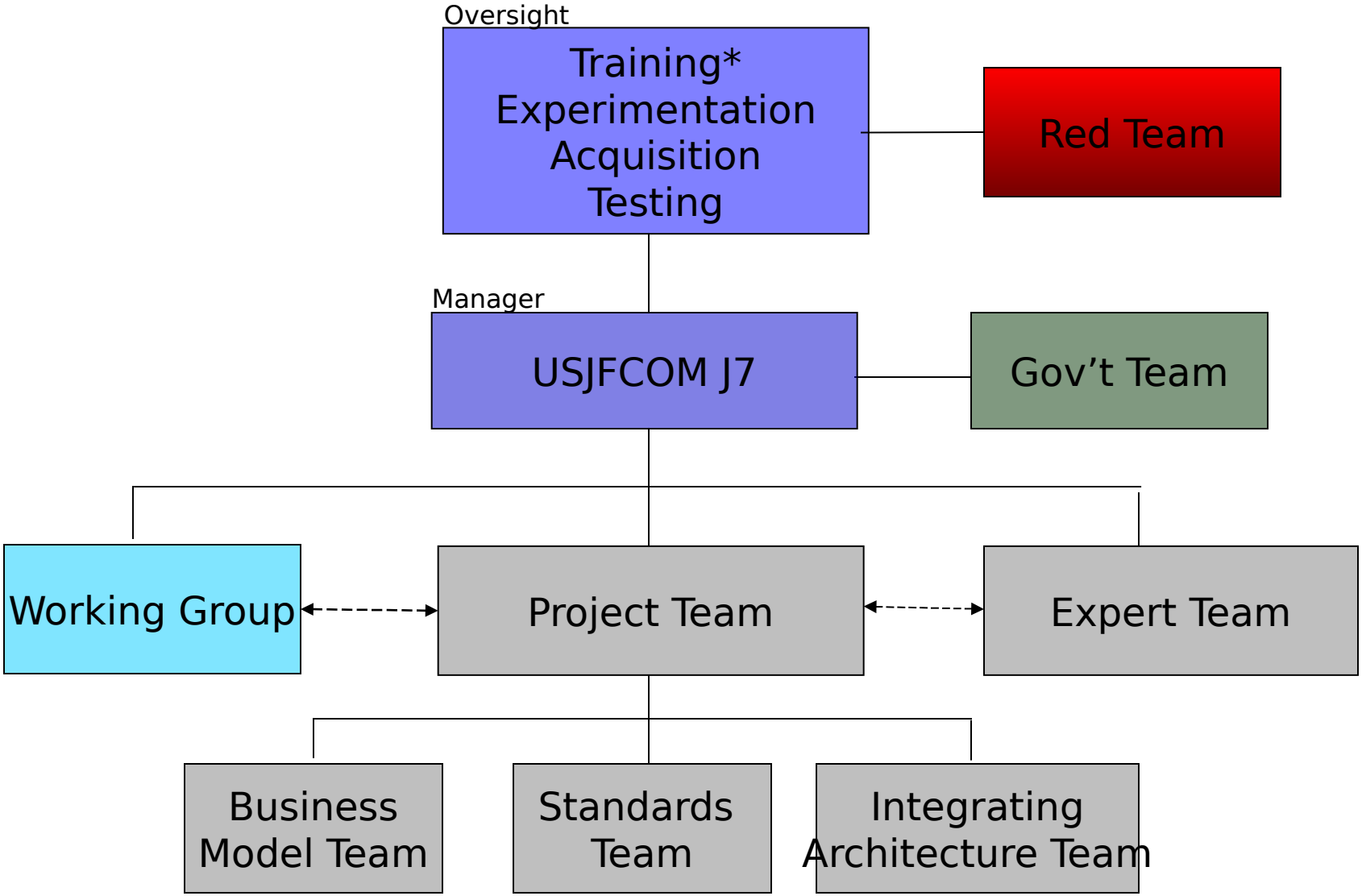


How do we move forward to best meet current and future need

Baseline Schedule



LVCAR Organization



Insights

- **Mixed architecture environments are a by-product of the simulations chosen for the application, not because of any inherent benefit to mixing architectures.**
- **When mixed architectures are necessary, point solutions to bridging the architectures do work, although they may be relatively inefficient.**
- **Architectural choices of how data is transferred between applications and application-level choices of what data will be passed have impacts on interoperability.**
- **Significant improvements in LVC interoperability can also be achieved via supporting data, tool, and process standards.**
- **There will be a need to recognize and account for longer-term trends (e.g., SOA) in the LVC "way ahead."**

Architectural Options

1. **Status Quo or "Do Nothing" – No architectural effort to unify or enhance the existing alternatives will be undertaken.** *Each existing architecture will evolve based on its own users' needs, and mixed-architecture events will continue to exist as currently achieved.*

2. **Actively Manage the Existing Architectures – Create standard inter-architecture integration solutions (effectively create an "architecture of architectures").** *Keep the current multiple architectures but invest in improving the construction / performance / integration of various gateways, translators, object models*, **and create processes and procedures to make inter-architecture integration "faster, easier, cheaper." Stand up an architecture management board (both policy and technical) to oversee all of the architectures to discourage divergence and encourage compatible evolution.**

3. **Convergence – Each of the existing architectures is evolving, some quickly, some slowly. Create policy and procedures, and provide small amounts of seed money, to** *encourage the architectures to converge with one another* **in X-year time frame (e.g., 10). When they become so similar in features and capabilities, engineer the merging of them into a single architecture. Requires an architecture management board (both policy and technical) to oversee all of the architectures.**

4. **Select One of the Existing Architectures – Of the existing architectures,** *choose the one that is the most promising for the long term* **DoD LVC community. Use policy and funding to throw the weight of the department behind the one chosen, make improvements where necessary, discourage the others, and eventually** *get to the situation where the chosen architecture is dominant*.

5. **Develop A New Architecture – With a better understanding of the broad DoD LVC requirements and the manifest lack of any of the current architectures to fully meet them any time in the future,** *create a new architecture from the best ideas of all the existing ones*, **and put the whole weight of the Department behind it.**

Management / Governance Considerations

- Alignment and establishment of relevant policy
- Allocation and influence of architecture related budgets
- Community Communication through papers, tutorials, liaison, …
- Product Support (technical assistance, cost sharing, LVC environment integration lab, …)
- Distribution of middleware/licenses and other tools
- Architecture requirements tracking, coordination, and arbitration
- Technical dispute tracking, coordination, and arbitration
- Participation in external standards bodies

What's Next?

- **Focus on the finish line:**
  - Gather additional metrics data for COA evaluation
  - Finalize COA recommendations
  - Detail "way ahead" activities and milestones

Contact Info

Project Manager

Ken Goad, kenneth.goad@jfcom.mil, (757) 203-5564

Project Engineer

Warren Bizub, warren.bizub@jfcom.mil, (757) 203-6969

Study Lead

Dr. Amy Henninger, ahenning@ida.org, (703) 845-6892

Questions

Backups

Terms of Reference

- **Interoperability:** **The ability of a system to provide information / services to and accept information / services from other systems, and to use the information / services so exchanged to enable them to operate effectively together.**
- **Integrating Architecture:** **A set of protocols, specifications, standards and/or middleware services that define and enable interoperability between LVC systems (e.g., TENA, HLA, DIS, CTIA).**

Architecture Requirements

- Create a distributed simulation, allow systems to join and resign; provide for initialization and destruction of the distributed simulation instance
- Support publish and subscribe information management
- Quality of service
- Support multiple message types
- Save and restore operation
- Region-based information management (filtering)
- Transfer of ownership
- Synchronize applications
- Object-oriented design
- Global event ordering
- Specification for Tools and Utilities

Future Requirements

- Quality of Service
- Fault Tolerance
- Information Assurance
- C4I System Integration
- Interface to GIG
- Load Balancing
- Gateways and Bridges
- Object Models